# MOBILE RADIO NETWORK

## Technical Field

[0001]    This invention relates to a mobile radio network and in particular to a mobile radio network that can be used with a wide variety of portable devices such as mobile computers, mobile phones, personal organizers, etc.

## Background of the Invention

[0002]    There is at present a divide in mobile computing between what is desirable and what is practical to implement.  The divide is caused by, inter-alia, constraints in size and power as well as the lack of reliable network connection.  There are a wide variety of mobile devices currently in use which would benefit from being part of a network which enables them to communicate with other devices.

[0003]    Because of the large variation in the types of devices which include data processing functions and which could be included in a network, these are very different communicational requirements.  There is, however, at the very least at base level of connectivity which is desirable for the most simple of functions such as monitoring the status of a particular device.

## Summary of the Invention

[0004]    Preferred embodiments of the present invention provide what we shall refer to as embedded networking.  This is concerned to provide a network that is so simple and small that it can be used by almost anything.  The network comprises a communications medium (typically radio), proximity sensors associated with devices, and a distributed database of resources.  It enables everyday objects to be able to communicate in a way which has not yet been achieved.  Many everyday electronic devices e.g. telephones, fax machines, photocopiers, electronic access controls, ATM machines and public information terminals already need a network in order to operate.  However, all can benefit from a common mechanism by which they are made aware of and can communicate with other things nearby.

[0005]    Preferred embodiments of the present invention provide a broad range of mobile and embedded objects which can detect by radio, other objects in their

environment capable of connecting to them, can connect to these with a low range ad hoc radio network, and can exchange any data necessary. To achieve this a basic node circuit has been developed which can provide connections directly to the objects in which they are embedded and which can communicate by radio to other similar nodes in their vicinity.

[0006]     Preferably, the nodes operate automatically to communicate with other nodes in their vicinity and to exchange any necessary data.

[0007]     The node is preferably provided on a single chip included in a device such as a mobile telephone, laptop computer, printer, etc.

## *Brief Description of the Drawings*

[0008]     The invention will now be described in detail by way of example with reference to the drawings in which:

[0009]     Figure 1(a) and (b) show two different types of communication systems which are provided by an embodiment of the present invention;

[0010]     Figure 2 shows a block diagram of a node for use in a mobile radio network embodying the present invention;

[0011]     Figure 3 shows a schematic diagram of the software used by the circuits of Figure 2;

[0012]     Figure 4 shows a datagram of the radio protocol used for communication between nodes in an embodiment of the invention;

[0013]     Figure 5 shows a radio boot protocol; and

[0014]     Figure 6 shows enquiries to and responses from an attribute store in an embodiment of the invention.

## *Detailed Description*

[0015]     To understand the usefulness of an embedded network we shall consider two different ways in which mobile devices can use services available from an embedded network. Figure 1(a) and (b) shows these schematically.

[0016]     Figure 1(a) shows a set of nodes or communication points which are able to move around by remain in range of each other. This is referred to as a cloud of devices

of the type which might be carried about by a person in luggage, in a vehicle or between a small group of people working in the same environment. Such devices can be made aware of each other via their nodes which can communicate by radio with other nodes and offer services to each other. They may be able to use each other's services sometimes for extended periods. For example, a personal data organizer may be authorized to use the mobile telephone of its user for e.g. sending and receiving messages by fax or E-mail.

[0017]    Figure 1(b) shows endpoints which include nodes and which move around occasionally coming within range of other nodes that provide services to which they have no special authorization. This is referred to as a nomadic node. The sort of services it might use are those that tell it about its environment e.g. position and local facilities, and those which might allow it to personalize another node by configuring it in a way that is suitable for a particular user. For example, a telephone may be pre-primed with a set of commonly dialed numbers when it detects a node owned by the person who has those commonly dialed numbers nearby.

[0018]    Thus, preferred embodiments of the present invention provide a system which is able to operate the systems outlined with reference to Figures 1(a) and (b).

[0019]    Radio technology is used for communications between nodes. This is because it possesses the characteristics needed for ad hoc, peer-to-peer communications in virtually all configurations and environments. The type of interaction which is required between nodes must be unrestricted, that is to say nodes must be able to communicate when in range even if they are being carried in a briefcase, coat pocket etc. Thus, infrared communication would not be appropriate in the general case because it requires a line of sight to be able to communicate.

[0020]    Systems embodying the present invention are decentralized. That is to say, every device must be able to independently describe itself to a sufficient level for it to be useful to others. This decentralized approach is used because knowledge about nearby objects which have nodes associated with them is more important than the knowledge of other devices which are not nearby. In particular, using such a system eliminates the need to contact and maintain a centralized database wherever a new node is encountered. That would require global connectivity.

[0021]     Figure 2 shows a block diagram of a node suitable for use in the mobile radio network. The core of this is a micro processor 2. This has three possible inputs, an analog input 4, a serial digital input 6, and a parallel digital input 8. In some special situations, it may be necessary to provide other inputs. Also, coupled to the microprocessor is a clock/alarm device 10. This wakes up the processor from low power sleep at programmed intervals.

[0022]     Radio controller logic 12 is connected to the micro processor 2 for controlling the provisions of signals to and from the microprocessor and this is used to control a radio transceiver 14 via which signals are sent to and received from other nodes which are in range. A 418MHZ FM transceiver is suitable for this purpose but other frequencies could be used.

[0023]     An example of the software scheme which could be loaded into the microprocessor 2 if Figure 2 is shown in Figure 3. A set of outputs corresponding to inputs 4, 6, 8 may be provided to send control signals to a device with which the node is associated. At its core is a kernel/scheduler 16 which is used to order all the functions of the node. This controls a radio driver 18 which is used by the radio controller logic 12. Memory management block 22 is used to organize temporary and permanent storage of data by the node. Timer software 24 controls the clock/alarm 10. A set of run time threads 26 are used to control data communications to and from the node once a nearby node has been detected, and an attribute store 28 which is used to store naming data and data describing the device with which the node is associated. This data is then used to identify and describe the device to other nearby nodes.

[0024]     Preferably the range of the radios used by nodes in the mobile radio network is constrained to about 5 meters. This makes the system into a proximity sensor as well as a radio network. This allows small low powered and cheap radios to be used. Because this range is small, greater re-use of the radio channel is available thereby increasing the aggregate bandwidth available for transmissions. Finally, it sets the system up to work at human ranges, that is to say it enables communication between objects that are within a human's immediate surroundings. In this case, those devices which are nearby to a person are things which can be connected to by the mobile radio network.

[0025]      Usually, if a device is close enough to be used by a person then it is one which should become part of the mobile radio network. Similarly, a few devices carried around by a person such as a mobile phone and a personal data organizer should be able to spontaneously inter-communicate when required.

[0026]      The radio protocol for the mobile radio network is selected to fit in with the ad hoc nature of the mobile radio network. That will be the protocol which provides short self-contained transactions between nodes rather than long sequences of patterns. Furthermore, some form of multicast communication between nodes must be possible to enable a device to communicate to all other devices in its vicinity that it is present and available for exchange of services. Also, some data about proximity and link quality between nearby devices is desirable.

[0027]      A suitable radio protocol for the system uses 4b6b encoded FM keying. A datagram showing the format of transmissions in the radio protocol format used is given in Figure 5. This comprises source and destination nodes, and addresses identifying the node from which data is being sent and for which it is intended. It also contains a protocol byte and payload length. The protocol byte is used internally in a node to determine how data is to be handled. The payload may contain up to 255 bytes of data being transmitted to another node. Because the data packet is fairly short, encoding and decoding for transmission is kept relatively simple. In addition, the use of small packets results in a better sharing of the radio channel between nodes since no single device will be transmitting for too long.

[0028]      Communication clouds of the type shown in Figure 1(a) will communicate with a well known group of nodes about which data is stored. Chance encounter transmissions of the type shown in Figure 1(b) are with transient groups of nodes with which a node associated with a particular device occasionally comes within range.

[0029]      As shown in Figure 5 the top two bits of the node address are used to identify the type of multicast group to which transmissions are to be made. Well known multicast addresses are used for pre-defined purposes and each of these are universally recognized by any node that may want to make use of it. Transient multicast allows dynamic creation of a multicast group, possibly for temporary collaboration or sharing between nodes. When a transient multicast group is created by a random 30 byte address

is nominated by one node for use by the group. Without any arbitration in the allocation of these addresses the statistical improbability of an addressing clash in both time and space is used to avoid conflicts. This address allocation strategy is used to serve the ad hoc nature of the mobile radio network.

[0030]      The components used in nodes are selected to make it easy to interface with the many different types of device that will use it. These will range from simple sensors which require analog, digital or serial input, to devices which may require more elaborate communication with the node. Each must have a common mechanism for use of the radio channel, particularly for device identification and description.

[0031]      Furthermore, it is preferable for it to be easy to configure a node for a particular task by booting it with application specific code. Furthermore, mobile nodes need to save power wherever possible. This means that the radio interface must be powered down most of the time. The two main components of the run time of a node are the kernel and the loader. The kernel 16 represents the environment that directs the operation of a node and provides drivers for the node's interfaces. Applications are downloaded into a node via the loader and are then run within the context of the kernel. The kernel exists in ROM in microprocessor 2. External RAM is provided to store specific applications, an application being a piece of code which is used to configure a node to perform specific tasks such as to act as an interface to a sensor connected to the node.

[0032]      The kernel 16 has a simple multi-threaded design. Within the kernel threads communicate by passing messages between one another. This is node via pairs of message queues managed by the scheduler. The kernel includes a number of system threads which are used to manage the internal components of the node. Alongside these, application or runtime threads 26 are used to make a node perform a particular task. The scheduler strategy employed dictates that any thread currently running must complete before any other thread can commence. Threads are identified by unique thread Ids. System threads are given well-p0known Ids. Application threads may be used to replace system threads thus upgrading the runtime environment and given the programmer control over a node.

[0033] A node is configured for a particular task by booting software into it through one of its external interfaces. This may be via the radio interface or via the hardware input shown in Figure 2. Once configuration is complete the node will retain its configuration indefinitely.

[0034] The radio booting procedure begins with a boot plea from the unconfigured node. This is issued on a well known multicast address, and contains details about the node such as its address, domain and version number. A node configured as a boot server may respond to the boot plea with a boot offer. A server does this after analysis of the boot plea in order to determine whether or not it has a suitable image to offer to that node. Upon receipt of a boot offer the node may decide to accept it and in doing so issues a boot request. After this the boot image is relayed to the node from the boot server node. This boot image may contain code and/or data. This procedure is illustrated in Figure 5.

[0035] Use of the radio boot protocol extends beyond merely configured a node. It represents a more general mechanism by which code and data can be passed between nodes. This is useful for dynamically upgrading the interface that a node offers to a particular service. Dynamic reconfiguration of a node has implications for power saving. There is a cost trade off between transmission and computation particularly in a low powered device. When transmission is much more costly than computation, it may be more efficient to configure a node with a particular decompression algorithm before sending data to it.

[0036] The attribute store 28 in the micro processor 2 is used to interpret data received over the radio interface of a node. Generally, it is necessary to map the source and destination addresses to particular types of devices. Additional information about the node may well be required in order to make any sense of data received from it. As the mobile radio network has no infrastructure base stations each node has to be individually responsible for describing itself to other nodes and for any other node to be able to interrogate it to determine what kind of services it requires or provides. The attribute store performs this function. The attribute store is a hierarchical directory structure of (name, value) pairs. It is available as a general resource within a node for presenting any kind of information to other nodes. It is a resource used both internally within the node

and externally as a presentation interface to other nodes. No type of information is stored in the attribute store since this could restrict the applications to which it can be put and will make its implementation more complicated. Standard names are used in the attribute store and these allow other nodes to discover the services available in their vicinity and to make use of them.

[0037]      The facilities are best described by way of example. A simple temperature sensor with a node interface intercepts information provided by the sensor and makes it available over the mobile radio network. Another node which comes within the range of this temperature sensing node will make the enquiries set out in Figure 6. The first enquiry is to get attribute name and the response is that the name is Temperature Sensor. The second enquiry asks for value associated with "temperature" and the response to this is 17. Finally, a watch request is sent. This causes the temperature sensing node to issue a notification of any change of temperature by passing the new temperature to the watching node. A watch will be cancelled after a predetermined period of time if the watching node does not renew it. Other more complex interrogation procedures are used for more complex devices.

[0038]      A node may be combined with a small low powered display to make the wireless portable information point. These can be provided within a building at pre-determined points so that a user may interrogate the node and the display will show information about past and present activity of the node.

[0039]      Nodes can be provided in many different forms. Some will be plugged into permanent power supplies to provide network access, environmental sensing, display services etc. Others will be included in portable devices and will be as small and lightweight as possible and would be powered by a coin sized power cell. These low power nodes will have to spend most of the time in sleep mode in which they can neither transmit nor receive radio signals. This leaves the problem of how low power nodes will ever discover each other's presence within their radio range if they are only active for short intervals.

[0040]      One way this can be done is by synchronizing their powered up time either by the use of internal clocks or an external time reference. Internal clocks tend to drift but this is something that can be overcome by periodic resynchronization. There may

also be problems with initial synchronization where two groups of differently synchronized nodes come within range of each other. External time references put extra hardware and complexity into every node and it is preferable to avoid these.

**[0041]** Another means by which the low power node may connect to another low power node is to use a low power RF detection circuit which responds to a transmission from another node to bring the lower power node out of sleep mode.

**[0042]** In order to enable nodes to identify each other, low power nodes are controlled to come out of sleep mode periodically and transmit a broadcast packet of data containing their unique ID. After the broadcast transmission, the low power node waits for a short period with its receiver on to detect if any other node is attempting to transmit back to it. Any node in the vicinity that receives the first broadcast packet has a short time to decide whether or not to respond with a transmission addressed to the low power node while the receiver is still on. The low power node, if it receives a response addressed to it during the short receive period, remains with its receiver on for a longer period, or responds with a transmission, and normal communications can be established.

**[0043]** Nodes in the vicinity may be able to listen for long periods for broadcast transmissions because they have greater power resources, or because they do this temporarily at particular times, or because they have been temporarily primed to listen by a special signal. This signal could be derived from a button that has been pressed to initiate rendezvous, or an environmental sensor such as a light detector or accelerometer, or some action by another node such as a specially coded radio transmission received by a very low power receiver or a burst of ultrasound or infra-red radiation. A typical time between the two transmissions of the broadcast packet will be 30 seconds but could be more or less frequent depending on the particular application.

**[0044]** Rendezvous is the term given to the means by which nodes initiate interaction. This may be done by some signaling communication on the radio channel between nodes. In order to describe how rendezvous may take place, we introduce the terms client and server to describe the entities involved in the rendezvous procedure.

**[0045]** The *server* provides as a service some piece of information, which may or may not be the result of a *request* from a *client*. The client request may contain information pertinent to the service being offered by the server – *parameters* of the

service, if you like. In this case, a more traditional query-response architecture is adopted whereby the client will issue a query containing the query's parameters. The query will be given an answer, the *response*, by the server.

[0046]    In such a configuration, the client and server must rendezvous before the parameters can be sent. This might be done by the server periodically "beaconing", sending a short radio packet to announce its presence. A client wishing to make use of the service will respond to the beacon with its request.

[0047]    Under some circumstances, the service offered by the server may be independent of any client parameters. In this case, it is possible to conserve the total power consumed within the system by introducing a "service in beacon" strategy. In this scheme, the server will periodically beacon, announcing its existence, and in addition, a service response will be held within the payload of the beacon. In doing this, the server precludes the need for any client to issue a query and wait for a response from the server, since the client has already received the response from the service it was looking for.

[0048]    Another power waving mechanism which might be adopted in the rendezvous procedure is that of *Proxy Rendezvous*. Consider the scenario where more than one client is waiting to make use of a nearby service. The normal way in which these clients would do this would be to have all the clients listening independently for the beacon from the server, and to all respond to this beacon with their specific queries.

[0049]    We can, however, optimize the total power consumption within a system of this kind by introducing a scheme of *Proxy Rendezvous*. Under this scheme, clients would "beacon" announcing their presence, and their interest in making use of a particular service, along with any query parameters they may have for this service. In hearing that another client is interested in making use of the same service as itself, the clients can rendezvous with each other, and nominate a client to "stay awake", waiting for the server's beacon. Other clients interested in this service can now sleep, and will re-rendezvous with the nominated client, who will have interacted with the server in their behalf.

[0050]    In a system where every node periodically broadcasts its unique ID the wake up time is followed by a brief interval for reception of data from other nodes.

Nodes can be arranged to listen to the available channel for longer periods at certain times to update them on the other nodes available in the vicinity.

[0051]    In another application, where a person has gathered together all the nodes that they wish to communicate with, only one of the nodes needs to be triggered into remaining active. This can then trigger all of the others as they make their periodic transmissions. Within a minute or so all nodes which are in range of the initial trigger should be able to intercommunicate directly and will have learned of each others' presence.

[0052]    Some nodes are associated with devices with large power supplies e.g. cellular phones, notebook computers, cars, and some will be connected to main electricity. These can remain active at all times and as with other nodes can be arranged to broadcast their IDs periodically e.g. every 30 seconds. These are referred to as listening nodes. When they detect a broadcast from another node they will look it up in a cache of recently detected nodes and if necessary will interrogate it to find out what it is and what services it offers and requires. If they have sufficiently recent information they will not respond. If a lower power node (A) is interrogated by a listening node (L) it can request a list of other recently seen nodes identified by their Ids. In addition, if node L overhears a service-in-beacon transmission, which may be marked as cacheable, it can store this service response alongside the node Id. This "cached" response would be marked as valid for a fixed time period. During this period, node L can re-transmit the cached response as a "proxy" for the service-in-beacon node. In this way, power consumption on the service-in-beacon node may be reduced. If it does not have knowledge of all the Ids it can immediately request details about them from the cache maintained by the listening node L. If the now power node (A) wishes to communicate with one of those recently seen nodes (B) it can choose to remain active for a pre-determined period of time e.g. 30 seconds and respond directly to the next broadcast from node B. It may even obtain a hint about how often and how recently that node broadcasts from the listening node L. If the node L cannot provide enough information about node B for A to make a decision then node A will remain active for long enough to establish direct contact with B and obtain sufficient details. Thus, the listening node is used purely as a hint device with communications being established directly if there is any interest.

This is important since the other nodes may no longer be in the vicinity, ie node B may have moved away from the listening node and may also have moved away from node A. Once the nodes are in contact A and B can power down for pre-determined periods and power up to transmit and listen for each other's presence. However, if it extends for too long the nodes will assume that they have lost contact and will return to their default state.

[0053] Other techniques may be adopted to save power progressively as a power source is consumed. A node could be arranged to detect a low battery condition and to make this available as a attribute to be read by a maintenance node. Levels of service offered by a node could be reduced as the node runs short of power. Finally, the node could lengthen the interval between its broadcast transmissions as its power runs out.

[0054] Higher data rates may be more attractive to nodes since the power per bit transmitted or received tends to be lower at higher bit rates. However, there will be a trade off with power consumption when a node was listening over a longer period since high speed radio receivers tend to consume more energy when receiving than transmitting. It may therefore be desirable to use a secondary radio for rendezvous purposes.

[0055] A particular set of nodes can be programmed for them to recognize each other and provide very specific services. For example, personal data organizer may recognize a telephone over which it is allowed to make calls or a printer to which it may send a document. Bringing the relevant device with its node into proximity with the phone or the printer will trigger these sorts of activities. Thus, it is the users action that causes things to happen. Once triggered connectivity between nodes may continue for some time.

[0056] If the embedded controllers in modern consumer devices were to be replaced with nodes, these devices could then communicate with each other when they are close enough to each other. Thus for example, any device that requires the time to be set (video cassette recorder, microwave oven, central heating controller etc.) would be able to obtain an update whenever one of them is set. This update could come from a service such as the radio data system, teletext, or the global positioning system that one of the nodes receives. Alternatively, it could be set by the user.

**[0057]** If the digital communications device is linked to a node then the communications medium may be used to control or interact with other nearby devices including nodes. For example, at home a modem on the telephone line equipped with a node could allow remote access to program a video cassette recorder, or central heating, or perhaps to control house lighting. A cellular phone equipped with a node would allow a portable computer to send and receive data messages must be being brought into sufficiently close proximity. Alternatively, calls could be diverted to a nearby wired telephone to save cellular band call costs. A network computer with a node could allow access to a nearby personal data organizer or display devices to exchange E-mail and important data files etc.

**[0058]** The mobile radio network could also be used for context aware applications. If a node is allocated to an individual it can be programmed with their personal preferences for common systems. For instance, telephone lists can be carried in the node and edited on any suitable computer. This could then be used to set the user interface on telephones equipped with nodes to set speed dialing programs.

**[0059]** Nodes could also be used in the working environment to act as beacons. For example, a node in a meeting room could request other nodes to be quiet when told that a meeting is in progress. This could then be used to suppress alarms, telephone calls, chimes etc. This node could also be linked to the meeting room booking facility to gain information on what the meeting was about. This could be recorded by participants' personal nodes along with the text names of other nodes at the meeting thus forming a record of their activities.

**[0060]** Nodes can also adopt a common format for location determination. Nodes that are known to be static e.g. the meeting room beacons, can be told their locations and can then make this available to passing nodes. Nodes attached to positioning systems (e.g. GPS) can make dynamic location information available in the same format in cars and aircraft. This data could then be used to trigger events when other nodes pass nearby. Personal nodes can record location against time and build up an accurate trace of where the node has been. Personal data organizer alarms for meetings etc can be related to location and will be given earlier if the organizer is far away but suppressed if the organizer is heading in the right direction or has arrived at the location. Devices such as

digital video recorders and cameras can tag their recorded information with locations discovered by an attached node.

[0061] Nodes can be used to augment direction signs with information about the routes to thousands of destinations. For example, road signs can carry electronic information about which exit to take to any destination in the UK that can be described by post code and street number. The user of a navigation device would enter the post code and number of their destination. The device would pass this to any nearby road signs which would respond with the appropriate exit to take and any ancillary information such as distance, expected journey time, road conditions etc this and other information could then be presented to the user on the display.

[0062] This of course is only one example. Nodes could be used to guide passengers on public transport, through buildings, around museums and galleries etc. Information could be made time dependent and could be updated swiftly in case of diversions or delays. It could also be personalized by the type of route e.g. scenic, fast, cheap, most stairs.

[0063] Another example, of how nodes could be used could be as a substitute for the presently used taped audio tours used in museums and outdoor tourists sites. With a node and a random access medium such as a mini disk the user could be free to vary the recommended itinerary. Routes could be customized to level of interest or length of time for the tour. If several people were taking the tour together their commentaries could be synchronized whenever they are together so that they are all looking at the same object at the same time. If nodes were provided on doorways and pathways and connected into a backbone network then lost tour members could be located. Finally, at the end of the tour a personalized printout containing the route followed and more details of the items where more time was spent could be automatically generated.

[0064] The above are only a few examples of the possible applications of this mobile radio network and the nodes required for it. Many other applications are possible.

[0065] The kernel has been designed with component modularity in mind. A thread and its associated state are encapsulated, and independent from the rest of the running kernel. Since threads only communicate by way of message queues, their interface to other threads is well constrained. This is very convenient from a perspective

of code mobility. It is very natural to be able to migrate a thread and its state from one node to another, via any of its external interfaces – in particular a serial line or the radio interface.

[0066]     By making code mobility an inherent part of the system design, we have encouraged a programming model which exploits the physical mobility of the system. One way in which nodes can interact is by sending a thread or threads from one device to another. These threads can perform some function on the remote node, log some data from a peripheral for example, and then return to the original host.

[0067]     In addition, code mobility could be used to minimize the total power consumption within the system. For example, if node A had a large amount of data to send over the radio interface to node B this may consume a large amount of energy because it would mean the radio transceiver being powered in both nodes for some considerable time. It may be more efficient for node B to send a thread containing a data compression algorithm to node A, and for node A to send compressed data over the radio interface, which would be decompressed at node B. By reducing the amount of data sent over the radio in this way, we may reduce the total energy consumed in the system.

[0068]     A sub-component of the Attribute Store is the Proximity Store. The Proximity Store contains a list of devices which have been interacted with, or overheard, on the radio channel recently. The structure of the Proximity Store is the same as that of the Attribute Store, and the interface that it presents to other components within a Piconet node is the same. The Proximity Store will retain the unique Id of any node that it has overheard recently, alongside as much information about that node as it has been able to determine. This might include a node's top-level service description, its beacon interval, and its transmission characteristics. Internal components of a node may make use of the Proximity Store as a trigger to indicate the arrival or departure of a particular node or type of node, as well as a hint as to the state of the local environment – which nodes it has spent a long time with, for example.

[0069]     In addition to storing a dynamic profile of the state of the local network environment, the Proximity store incorporates a component which maintains an historical record of the contents of the Proximity Store over time. This is used by internal

components of the node to build up a broader view of the traffic profile of the radio network.